

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2004-326743

(43)Date of publication of application : 18.11.2004

(51)Int.Cl. G06F 9/46

(21)Application number : 2004-087908 (71)Applicant : INTERNATL BUSINESS MACH CORP <IBM>

(22)Date of filing : 24.03.2004 (72)Inventor : ARMSTRONG WILLIAM JOSEPH
LARSON DAVID ANTHONY
NAYAR NARESH

(30)Priority

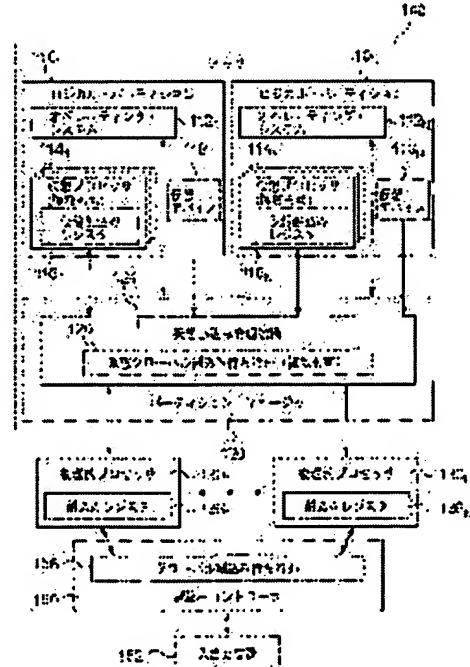
Priority number : 2003 422513 Priority date : 24.04.2003 Priority country : US

(54) VIRTUALIZATION OF GLOBAL INTERRUPT QUEUE

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a method, system, and a program article of manufacture for processing virtual interrupts in a logically partitioned system.

SOLUTION: An intelligent virtual global interrupt queue (virtual GIQ) that may be associated with a plurality of virtual processors running in a logical partition may be utilized. Upon receiving a virtual interrupt, the virtual GIQ may examine the operating states of the associated virtual processors. In an effort to ensure the virtual interrupt is processed as quickly as possible, the virtual GIQ may present the virtual interrupt to one of the associated virtual processors determined to be in an operating state best suited for processing the virtual interrupt.



BEST AVAILABLE COPY

LEGAL STATUS

[Date of request for examination] 24.03.2004

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2004-326743

(P2004-326743A)

(43) 公開日 平成16年11月18日 (2004. 11. 18)

(51) Int.Cl.⁷
G06F 9/46F I
G06F 9/46 350
G06F 9/46 320Zテーマコード (参考)
5B098

審査請求 有 請求項の数 21 O.L. (全 16 頁)

(21) 出願番号 特願2004-87908 (P2004-87908)
 (22) 出願日 平成16年3月24日 (2004. 3. 24)
 (31) 優先権主張番号 10/422513
 (32) 優先日 平成15年4月24日 (2003. 4. 24)
 (33) 優先権主張国 米国 (US)

(特許庁注: 以下のものは登録商標)

1. Linux

(71) 出願人 390009531
 インターナショナル・ビジネス・マシーンズ・コーポレーション
 INTERNATIONAL BUSINESS MACHINES CORPORATION
 アメリカ合衆国 10504 ニューヨーク
 州 アーモンク ニュー オーチャード
 ロード
 (74) 代理人 100086243
 弁理士 坂口 博
 (74) 代理人 100091568
 弁理士 市位 嘉宏
 (74) 代理人 100108501
 弁理士 上野 剛史

最終頁に続く

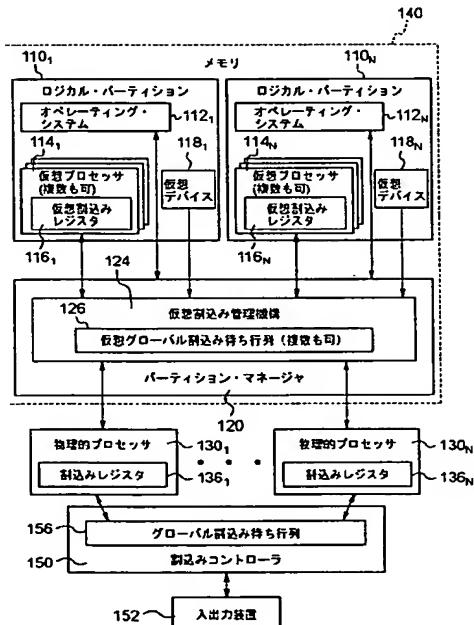
(54) 【発明の名称】グローバル割込み待ち行列の仮想化

(57) 【要約】

【課題】ロジカル・パーティション・システム内で仮想割込みを処理するための方法、システム、およびプログラム製品を提供すること。

【解決手段】1つのロジカル・パーティション内で動作する複数の仮想プロセッサに関連する可能性のあるインテリジェント仮想グローバル割込み待ち行列(仮想G I Q)を使用することができる。仮想割込みを受信すると、仮想G I Qは、関連仮想プロセッサの動作状態を検査することができる。仮想割込みができるだけ迅速に処理されることを保証しようと努力して、仮想G I Qは、仮想割込みを処理するのに最も適した動作状態にあると判定された関連仮想プロセッサの1つに仮想割込みを提示することができる。

【選択図】 図1



【特許請求の範囲】**【請求項1】**

仮想割込みを処理するための方法において、
複数の仮想プロセッサのうちの少なくとも1つの動作状態に基づいて前記仮想割込みを処理するために前記複数の仮想プロセッサから1つを選択するステップと、
前記選択した仮想プロセッサに前記仮想割込みを提示するステップとを有する方法。

【請求項2】

前記仮想割込みを処理するために前記仮想プロセッサから1つを選択するステップが、
前記仮想プロセッサのうちの少なくとも1つの前記動作状態を検査して、前記少なくとも1つの仮想プロセッサが1つまたは複数の基準を満たしているかどうかを判定するステップを有する、請求項1に記載の方法。

10

【請求項3】

前記基準のうちの1つが、前記少なくとも1つの仮想プロセッサが現在実行中であるかどうかを有する、請求項2に記載の方法。

【請求項4】

前記仮想割込みのソースを決定するステップをさらに有し、
前記基準のうちの1つが、前記少なくとも1つの仮想プロセッサが前記仮想割込みの前記ソースをマスクしているかどうかを有する、請求項2に記載の方法。

20

【請求項5】

前記少なくとも1つの仮想プロセッサが実行中であり、前記仮想割込みソースがマスクされていない場合に、前記少なくとも1つの仮想プロセッサに前記仮想割込みを提示するステップをさらに有する、請求項4に記載の方法。

【請求項6】

前記少なくとも1つの仮想プロセッサに関連する仮想割込みレジスタに基づいて前記少なくとも1つの仮想プロセッサの前記仮想割込みソースがマスクされているかどうかを判定するステップであって、前記仮想割込みレジスタが前記少なくとも1つの仮想プロセッサの現行割込み優先順位レベルを示すステップをさらに有する、請求項5に記載の方法。

30

【請求項7】

仮想割込みを提示するために複数の仮想プロセッサから1つを選択するための方法において、

前記仮想プロセッサのうちのいずれかが現在実行中であるかどうかを判定するステップと、

実行中である場合にいずれかの実行中の仮想プロセッサの前記仮想割込みのソースがマスクされていないかどうかを判定するステップと、

マスクされている場合に現在実行中であり、前記仮想割込みソースがマスクされていないと判定された仮想プロセッサに前記仮想割込みを提示するステップとを有する方法。

【請求項8】

前記仮想プロセッサのいずれも現在実行中ではなく、前記仮想割込みのソースがマスクされていない場合に、

40

前記仮想プロセッサのうちのいずれかが現在実行中であるかどうかを判定するステップと、

実行中である場合に現在実行中であると判定された仮想プロセッサに前記仮想割込みを提示するステップとをさらに有する、請求項7に記載の方法。

【請求項9】

前記仮想プロセッサのいずれも現在実行中ではない場合に、

前記仮想プロセッサのうちのいずれかの前記仮想割込みの前記ソースがマスクされていないかどうかを判定するステップと、

マスクされている場合に前記仮想割込みの前記ソースがマスクされていないと判定された仮想プロセッサに前記仮想割込みを提示するステップとをさらに有する、請求項8に記載の方法。

50

【請求項 10】

前記仮想プロセッサのいずれも現在実行中ではなく、前記仮想プロセッサのそれぞれの前記割込みの前記ソースがマスクされている場合にのみ、現在実行中ではなく、前記仮想割込みの前記ソースがマスクされている仮想プロセッサに前記仮想割込みを提示するステップをさらに有する、請求項 9 に記載の方法。

【請求項 11】

仮想割込みを処理するために複数の仮想プロセッサから 1 つを選択するためのコンピュータ・プログラムにおいて、
プロセッサによって実行されると、

前記仮想プロセッサの動作状態に基づいて前記仮想割込みを処理するために前記複数の仮想プロセッサから 1 つを選択するステップと、前記選択した仮想プロセッサに前記仮想割込みを提示するステップとを有する動作を実行する、コンピュータ・プログラム。

【請求項 12】

前記仮想割込みを処理するために前記仮想プロセッサから 1 つを選択するステップが、
前記仮想プロセッサのうちの少なくとも 1 つの前記動作状態を検査して、前記少なくとも 1 つの仮想プロセッサが 1 つまたは複数の基準を満たしているかどうかを判定するステップを有する、請求項 1 1 に記載のコンピュータ・プログラム。

【請求項 13】

前記基準のうちの 1 つが、前記少なくとも 1 つの仮想プロセッサが現在実行中であるかどうかを有する、請求項 1 2 に記載のコンピュータ・プログラム。

10

20

【請求項 14】

前記動作が、
前記仮想割込みのソースを決定するステップをさらに有し、
前記基準のうちの 1 つが、前記少なくとも 1 つの仮想プロセッサが前記仮想割込みの前記ソースをマスクしているかどうかを有する、請求項 1 3 に記載のコンピュータ・プログラム。

【請求項 15】

前記動作が、前記少なくとも 1 つの仮想プロセッサが実行中であり、前記仮想割込みソースがマスクされていない場合に、前記少なくとも 1 つの仮想プロセッサに前記仮想割込みを提示するステップをさらに有し、前記基準のうちの 1 つが、前記少なくとも 1 つの仮想プロセッサが現在実行中であるかどうかを有する、請求項 1 2 に記載のコンピュータ・プログラム。

30

【請求項 16】

少なくとも 1 つのロジカル・パーティションと、
前記ロジカル・パーティションに関連する複数の仮想プロセッサと、
前記ロジカル・パーティションをターゲットにする仮想割込みを受信し、前記仮想プロセッサのうちの少なくとも 1 つの動作状態に基づいて前記仮想割込みを処理するために前記仮想プロセッサから 1 つを選択し、前記選択した仮想プロセッサに前記仮想割込みを提示するように構成される仮想グローバル割込み待ち行列とを有するロジカル・パーティション・コンピュータ・システム。

40

【請求項 17】

現在実行中の仮想プロセッサがまったくない場合にのみ、現在実行中ではない仮想プロセッサに仮想割込みを提示するように前記仮想グローバル割込み待ち行列が構成される、請求項 1 6 に記載のロジカル・パーティション・コンピュータ・システム。

【請求項 18】

複数の仮想プロセッサが実行中である場合に、
前記仮想割込みのソースをマスクしていない実行中の仮想プロセッサがまったくない場合にのみ、前記仮想割込みの前記ソースをマスクしている実行中の仮想プロセッサに割込みを提示するように前記仮想グローバル割込み待ち行列が構成される、請求項 1 7 に記載のロジカル・パーティション・コンピュータ・システム。

50

【請求項 19】

仮想割込みを生成可能な 1 つまたは複数の仮想デバイスをさらに有する、請求項 16 に記載のロジカル・パーティション・コンピュータ・システム。

【請求項 20】

前記仮想グローバル割込み待ち行列が、複数のロジカル・パーティション間で共用されるハードウェア・グローバル割込み待ち行列を仮想化する、請求項 16 に記載のロジカル・パーティション・コンピュータ・システム。

【請求項 21】

複数の関連仮想プロセッサを有する少なくとも 1 つの追加のロジカル・パーティションと、

10

前記少なくとも 1 つの追加のロジカル・パーティションに関連する少なくとも 1 つの追加の仮想グローバル割込み待ち行列とを有する、請求項 16 に記載のロジカル・パーティション・コンピュータ・システム。

【発明の詳細な説明】**【技術分野】****【0001】**

本発明は一般にロジカル・パーティション・コンピュータ・システムに関し、より詳細にはロジカル・パーティションをターゲットにする仮想割込みの処理に関する。

【背景技術】**【0002】**

コンピューティング環境における並列処理は、一般に、複数のコンピューティング・タスクを並列に実行することを指す。伝統的に、並列処理は複数のコンピュータ・システムを必要とし、各コンピュータ・システムのリソースは特定のタスクに専用であるかまたは共通タスクの一部分を実行するために割り振られていた。しかし、最近のコンピュータ・ハードウェアおよびソフトウェア技術の進歩の結果、システム・リソースを種々のタスクにロジカル・パーティション化することにより、單一コンピュータ・システムは非常に複雑な並列処理が可能になっている。ロジカル・パーティション・コンピュータ・システムでは、使用可能なシステム・リソースが複数のロジカル・パーティション間に割り振られ、それぞれが他のものとは無関係に動作するように見えるように設計されている。ロジカル・パーティション間のリソースの割振りの管理は通常、一般にパーティション・マネージャと呼ばれるソフトウェア・コンポーネントの層を介して実施される。

30

【0003】

パーティション・マネージャの目的は、通常、ほとんどまたはまったく修正せずに専用コンピュータ・システム上で動作するように開発されたソフトウェア（たとえば、オペレーティング・システムおよびオペレーティング・システム固有アプリケーション）を各ロジカル・パーティションが独立して実行できるようにすることである。たとえば、1つのロジカル・パーティションが IBM の OS/400 などの第 1 のオペレーティング・システムを実行し、第 2 のロジカル・パーティションが IBM の AIX などの第 2 のオペレーティング・システムを実行し、第 3 のロジカル・パーティションが Linux などの第 3 のオペレーティング・システムを実行している場合もある。同じコンピュータ・システム上で複数のオペレーティング・システムを実行する能力を提供することにより、ロジカル・パーティション・システムは、それに関してアプリケーション・プログラムが作成されたオペレーティング・システムをほとんどまたはまったく考慮せずにユーザの要求に最も適したアプリケーション・プログラムを選択する際により大きい自由度をユーザに提供することができる。

40

【0004】

パーティション・マネージャは通常、ロジカル・パーティションの見地から見ると、対応するハードウェア・コンポーネントと同一の方式で動作する 1 組の仮想リソース（ソフトウェア・コンポーネント）を各ロジカル・パーティションに提示することにより、各ロジカル・パーティションがソフトウェアを独立して実行できるようにするという目的を達

50

成する。換言すれば、パーティション・マネージャは、各ロジカル・パーティションが実質的にそれ自体の1組の仮想リソースを備えた独立仮想コンピュータ・システム（または仮想計算機）として動作できるようにすることができる。

【0005】

各ロジカル・パーティションに提示される仮想リソースは1つまたは複数の仮想プロセッサを含むことができる。各パーティション用の仮想プロセッサは、そのパーティションに割り当てられた1つまたは複数の物理的プロセッサの処理リソースを共用することができる。他のタイプの仮想デバイスのサポートを提供するためならびに複数のオペレーティング・システムが1つのパーティション化システム上のプロセッサを共用できるようにするため、仮想プロセッサは仮想割込みを処理できる可能性がある。換言すれば、実際のハードウェア装置が物理的プロセッサ上で割込みを発生する場合があるように、仮想割込みが仮想デバイスによって生成される場合もあれば、仮想割込みを受信した仮想プロセッサを実行しているパーティションとは異なるパーティションからリルート(reroute)された実際のハードウェア割込みの結果として生成される場合もある。仮想割込みを作成するという概念は、2003年3月31日に出願され、「Apparatus and Method for Virtualizing Interrupts in a LogicallyPartitioned Computer System」という名称で、本出願人が所有する特許出願第10/403158号に詳細に記載されている。

10

【0006】

ロジカル・パーティション・コンピュータ・システムなど、複数の物理的プロセッサを備えたコンピュータ・システムは、システム内の物理的プロセッサのうちのどの物理的プロセッサが入出力装置による割込みのターゲットになりうるかを本質的に指定する割込み管理機構（一般に、グローバル割込み待ち行列という）を使用することができる。割込みを受信すると、グローバル割込み待ち行列ハードウェア（および／またはソフトウェア）は、割込みを提示するために関連物理的プロセッサのうちの1つを選択し、それにより、割込み処理のオーバヘッドを複数のプロセッサ間に分散することができる。一部の例では、グローバル割込み待ち行列は、複数の物理的プロセッサから選択した1つであって、他のものより割込みの処理に有利な実行状態にある物理的プロセッサに割込みを提示するように設計することができる。たとえば、他の物理的プロセッサは、他のパーティション・タスクの処理で使用中である場合もあれば、すでに他の割込み（たとえば、優先順位レベルが高いもの）を処理している場合もある。

20

【0007】

仮想プロセッサは物理的プロセッサと同様の挙動を示すように設計されているので、1つのロジカル・パーティションに割り当てられた1組の仮想プロセッサも他のものより仮想割込みの処理に有利な実行状態を有することができる。そのロジカル・パーティションをターゲットにする割込みができるだけ早く処理されることを保証するために、最も有利な実行状態にある仮想プロセッサに割込みが提示されることが望ましいだろう。しかし、仮想割込みを受信して、それを処理するために最も有利な動作状態にある仮想プロセッサにそれを経路指定するための機構（たとえば、グローバル割込み待ち行列の機能性と類似した機能性を備えたもの）は、現在、まったく存在しない。

30

【特許文献1】2003年3月31日に出願され、「Apparatus and Method for Virtualizing Interrupts in a LogicallyPartitioned Computer System」という名称で、本出願人が所有する特許出願第10/403158号

40

【特許文献2】2001年4月19日に出願され、「Method and Apparatus for Allocating Processor Resources in a Logically Partitioned Computer System」という名称で、本出願人が所有する特許出願第09/838057号

【発明の開示】

【発明が解決しようとする課題】

【0008】

したがって、たとえば、仮想割込みを処理するために有利な動作状態にあると判定された仮想プロセッサに仮想割込みを経路指定するように構成された仮想グローバル割込み待

50

ち行列を実現することにより、仮想割込みを処理するための改良された方法およびシステムが必要である。

【課題を解決するための手段】

【0009】

本発明は一般に、仮想割込みを処理するための方法、プログラム製品、およびシステムを対象とする。

【0010】

一実施形態は、仮想割込みを処理するための方法を提供する。この方法は一般に、複数の仮想プロセッサのうちの少なくとも1つの動作状態に基づいて仮想割込みを処理するために複数の仮想プロセッサから1つを選択するステップと、選択した仮想プロセッサに仮想割込みを提示するステップとを含む。

10

【0011】

他の実施形態は、仮想割込みを提示するために複数の仮想プロセッサから1つを選択するための方法を提供する。この方法は一般に、仮想プロセッサのうちのいずれかが現在実行中であるかどうかを判定するステップと、実行中である場合にいずれかの実行中の仮想プロセッサの仮想割込みのソースがマスクされていないかどうかを判定するステップと、マスクされている場合に現在実行中であり、その仮想割込みソースがマスクされていないと判定された仮想プロセッサに仮想割込みを提示するステップとを含む。

【0012】

他の実施形態は、仮想割込みを処理するために複数の仮想プロセッサから1つを選択するためのプログラムを含むコンピュータ・プログラムを提供する。プロセッサによって実行されると、このプログラムは、一般に、仮想プロセッサの動作状態に基づいて仮想割込みを処理するために複数の仮想プロセッサから1つを選択するステップと、選択した仮想プロセッサに仮想割込みを提示するステップとを含む動作を実行する。

20

【0013】

他の実施形態は、少なくとも1つのロジカル・パーティションと、そのロジカル・パーティションに関連する複数の仮想プロセッサと、仮想グローバル割込み待ち行列とを含むロジカル・パーティション・コンピュータ・システムを提供する。仮想グローバル割込み待ち行列は一般に、そのロジカル・パーティションをターゲットにする仮想割込みを受信し、その仮想プロセッサのうちの少なくとも1つの動作状態に基づいて仮想割込みを処理するために複数の仮想プロセッサから1つを選択し、選択した仮想プロセッサに仮想割込みを提示するように構成される。

30

【0014】

添付図面に例示した本発明の諸実施形態を参照することにより、上記で列挙した本発明の特徴を達成する方法を詳細に理解でき、上記で簡単に要約した本発明のより詳細な説明が可能になるだろう。

【0015】

しかし、添付図面は本発明の典型的な実施形態のみを例示し、したがって、その範囲を限定するものと見なすべきではなく、本発明は他の同様に効果的な実施形態を認めることができることに留意されたい。

40

【発明を実施するための最良の形態】

【0016】

本発明は一般に、ロジカル・パーティション・システム内で仮想割込みを処理するための方法、システム、および製品を対象とする。実施形態では、1つのロジカル・パーティション内で動作する複数の仮想プロセッサに関連する可能性のあるインテリジェント仮想グローバル割込み待ち行列（仮想G I Q）を提供し使用する。仮想割込みを受信すると、仮想G I Qは、関連仮想プロセッサの動作状態を検査することができる。仮想割込みができるだけ迅速に処理されることを保証しようと努力して、仮想G I Qは、仮想割込みを処理するのに最も適した動作状態にある関連仮想プロセッサの1つに仮想割込みを提示することができる。

50

【0017】

本発明の一実施形態は、たとえば図1に示し後述するロジカル・パーティション・コンピュータ・システム100などのコンピュータ・システムとともに使用するためのプログラム記録媒体として実現される。このプログラム記録媒体のプログラム（複数も可）は、諸実施形態（本明細書に記載する方法を含む）の諸機能を定義するものであり、様々な信号運搬媒体上に含めることができる。例示的な信号運搬媒体としては、（i）書き込み不能記憶媒体（たとえば、CD-ROMドライブによって読み取り可能なCD-ROMディスクなどのコンピュータ内の読み取り専用メモリ・デバイス）上に永続的に記憶された情報、（ii）書き込み可能記憶媒体（たとえば、ディスクケット・ドライブ内のフレキシブル・ディスクまたはハードディスク・ドライブ）上に記憶された可変情報、（iii）ワイヤレス通信およびインターネットを含むコンピュータ・ネットワークまたは電話網によるなどの通信媒体によってコンピュータに伝達される情報を含むが、これらに限定されない。

10

【0018】

一般に、本発明の諸実施形態を実現するために実行されるルーチンは、たとえば、図1に示すロジカル・パーティション・コンピュータ・システム100のパーティション・マネージャ120の仮想割込み管理機構124の1つのコンポーネントにすることができる仮想グローバル割込み待ち行列126として実施されるファームウェアを含む、オペレーティング・システムの一部または特定のアプリケーション、コンポーネント、プログラム、モジュール、オブジェクト、または命令シーケンスにすることができる。本発明のソフトウェアは通常、ネイティブ・コンピュータによって機械可読フォーマットに、したがって実行可能命令に変換される多数の命令から構成される。また、プログラムは、プログラムにとってローカルに常駐するか、メモリ内または記憶装置上で検出される変数およびデータ構造から構成される。加えて、以下に記載する様々なプログラムは、本発明の特定の実施形態で実現されるアプリケーションに基づいて識別することができる。しかし、以下に示す特定の名称は単に便宜的に使用するものであり、したがって、本発明はこのような名称によって識別または暗示される特定のアプリケーションのみで使用することに限定するべきではないことを理解されたい。

20

【0019】

例示的なロジカル・パーティション・システム

図1は、1つまたは複数のロジカル・パーティション110（任意の数N個のロジカル・パーティション110をサポートできることを表すためにロジカル・パーティション110₁～110_Nとして示す）を有するロジカル・パーティション・コンピュータ・システム100を示している。パーティション・マネージャ120は一般に、ロジカル・パーティション110の作成および削除を制御することができる。各ロジカル・パーティション110は、1つまたは複数の仮想プロセッサ114（1つまたは複数の物理的プロセッサ130を共有することができるもの）上で動作する関連オペレーティング・システム112を有することができる。仮想プロセッサの概念は、2001年4月19日に出願され、「Method and Apparatus for Allocating Processor Resources in a logically Partitioned Computer System」という名称で、本出願人が所有する特許出願第09/838057号に詳細に記載されている。

30

【0020】

コンピュータ・システム100は、ネットワーク・サーバ、メインフレーム・コンピュータなど、ロジカル・パーティションをサポート可能な適当なタイプのコンピュータ・システムにることができる。一実施形態では、コンピュータ・システム100は、ニューヨーク州アーモンクのIBM社から入手可能なeServer iSeriesコンピュータ・システムである。コンピュータ・システム100は一般に、メモリ140と結合された1つまたは複数の物理的プロセッサ130を含む。システム・プロセッサ130は、適当な割振り配置に応じてロジカル・パーティション110間で割り振ることができる。

40

【0021】

たとえば、各ロジカル・パーティション110は、システム・プロセッサ130のうち

50

、それ自体に専用の1つまたは複数のプロセッサを有する場合もあれば、システム・プロセッサ130のうちの1つまたは複数を1つまたは複数の他のロジカル・パーティション110と共有する場合もある。各システム・プロセッサ130は、ロジカル・パーティション110の仮想プロセッサ114のうちの1つまたは複数を実行することができる。ロジカル・パーティション110間におけるシステム・プロセッサ130、メモリ140、ならびに入出力装置152（広範囲の様々な入出力装置ならびに入出力インターフェース、記憶インターフェース、ネットワーク・インターフェースなどのインターフェース装置を含むことができる）などの様々な他のリソースの割振りは、パーティション・マネージャ120によって制御することができる。

【0022】

10

図示の通り、システム100は、入出力装置152が生成した割込みを受信し、グローバル割込み待ち行列156が指定するように物理的プロセッサ130のうちの1つまたは複数にその割込みを経路指定するように構成された割込みコントローラ150を含むことができる。パーティション・マネージャ120は、一般に（たとえば、グローバル割込み待ち行列156を介して提示される入出力割込みの結果として）物理的プロセッサ130からの割込みを受信し、処理のためにロジカル・パーティションの1つの仮想プロセッサ114にその割込みを経路指定するように構成された仮想割込み管理機構124を含むことができる。

【0023】

20

仮想割込み管理機構124は、その割込みがどのロジカル・パーティション110をターゲットにしているかを決定し、このような詳細を物理的プロセッサ130から隠すことができる。たとえば、仮想割込み管理機構124は、（たとえば、物理的プロセッサ130の割込みレジスタ136内で識別された通り）割込みソースに基づいてその割込みがどのロジカル・パーティション110をターゲットにしているかを決定することができる。割込みを処理するために、仮想割込みは、ターゲットのロジカル・パーティション110上で動作する仮想プロセッサ114の1つに提示することができる。また、仮想割込み管理機構124は、1つまたは複数の仮想デバイス118上で発生するイベントに応答して仮想割込みを生成することもできる。

【0024】

30

仮想グローバル割込み待ち行列（仮想G I Q）

いずれの場合も、仮想割込みを提示するために使用可能な仮想プロセッサ114は、ターゲットのロジカル・パーティション110に関連する仮想G I Q126によって指定することができる。ただし、ロジカル・パーティション110は仮想G I Q126に割り当てられていない仮想プロセッサ114上で動作している可能性があり、このような仮想プロセッサ114が仮想G I Q126を介して仮想割込みを受信しないことを暗示していることに留意されたい。しかし、その場合でもこのような仮想プロセッサ114は、特にそれらをターゲットとする仮想割込みのうち、G I Q126により送信されない仮想割込みを受信することができる。図2および図3に示すように、各ロジカル・パーティション110はそれ自体の仮想G I Q126（たとえば、仮想G I Q126₁～126_N）を備えることができる。実現例に応じて、図2に示すように各仮想G I Q126が対応するロジカル・パーティション110に専用のハードウェアG I Q156を仮想化する場合もあれば、図3に示すようにすべての仮想G I Q126が複数のロジカル・パーティション110間で共用されるハードウェアG I Q156を共同で仮想化する場合もある。

40

【0025】

いずれの場合も、仮想G I Q126は一般に、それぞれの動作状態に基づいてその関連仮想プロセッサ114のうちのどの仮想プロセッサが仮想割込みを処理するのに最も適しているかを決定するように構成することができる。この決定を行なう際に、仮想G I Q126は、仮想割込みの処理に有利な1つまたは複数の基準を満たしているかどうかを立証するために、その仮想プロセッサ114の動作状態を検査することができる。どの仮想プロセッサ114が仮想割込みを処理するのに最も適しているかを決定するために多種多様な

50

動作状態基準を適用することができるが、その特定の詳細は異なる実施形態ごとに様々になる可能性がある。

【0026】

仮想割込みの提示のために仮想プロセッサを選択するときに適用可能な適当な基準の2つの例は、その仮想プロセッサが現在実行中であるかどうかと、その仮想プロセッサが仮想割込みのソースをマスクしているかどうかであり、いずれも仮想プロセッサが仮想割込みを処理できる速度に影響を及ぼす可能性がある。たとえば、第1の仮想プロセッサが現在実行中ではない場合（たとえば、第1のプロセッサが優先使用または放棄され、第2の仮想プロセッサが実行中である場合）、第1の仮想プロセッサが仮想割込みを処理するためには、（たとえば、その実行状態が保管されるなど）第2の仮想プロセッサが先に優先使用されなければならず、第2の仮想プロセッサがディスパッチされるのを待つ間に遅延が発生することになるだろう。仮想プロセッサが実行中であるかどうかは、たとえば、その仮想プロセッサに関連する制御ロック（図示せず）内のレジスタを介して判定することができる。

【0027】

仮想割込みを処理する際の遅延は、仮想割込みのソースをマスクした仮想プロセッサに対して仮想割込みを送信する場合にも発生する可能性がある。割込みマスクに関する的確な技法は異なるプロセッサごとに様々になる可能性がある。一般的な技法の1つは、現行割込み優先順位レベルを維持管理し、それにより、受信した割込みのうち、より高い割込み優先順位レベルを有する割込みのみを処理することである。割込み優先順位レベルによる割込みマスクの概念は、それぞれ「05」および「FF」（たとえば、16進数の「FF」 = 10進数の「255」）という現行割込み優先順位レベルを有する2つの仮想プロセッサ VP1 114₁ および VP2 114₂ を示す図4に示されており、VP2 がいかなる割込みもマスクしないように「FF」は可能な範囲で最低の優先順位レベルである。この例の場合、仮想プロセッサ VP1 と VP2 がいずれも実行中であると想定する。「10」という優先順位レベルを有する仮想割込みを受信すると、仮想 GIQ は、VP1 に提示された場合、VP1 が「10」より大きい優先順位を有する現在保留中のすべての割込みを処理してしまうまで、その仮想割込みが処理されないだろう（たとえば、VP1 の割込み待ち行列内で保留になるだろう）と判断する可能性がある。これに反して、「FF」という現行割込み優先順位レベルを有しているので、VP2 は仮想割込みを直ちに処理することができるだろう。したがって、仮想 GIQ 126 は VP2 に仮想割込みを提示する。

【0028】

仮想割込みを経路指定するための例示的な方法

図3は、たとえば、仮想割込みを処理するのに最も適した仮想プロセッサを選択するために仮想 GIQ 126 が実行可能な例示的な動作300を示す流れ図である。動作300は、仮想割込みを受信することにより、ステップ302から始まる。前述の通り、仮想割込みは、パーティション上の仮想デバイス118によるか（たとえば、ソフトウェア割込み）またはハードウェア割込みの結果として（たとえば、入出力装置152から）生成することができる。いずれの場合も、仮想 GIQ 126 は、たとえば、前述の通り、どの VP が1組の基準を満たしているかを立証しようと努力して各 VP の動作状態を検査することにより、その関連 VP のうち、どの VP が割込みを処理するのに最も適しているかを決定しようと試みる。

【0029】

したがって、ステップ304で仮想 GIQ は、各 VP の動作状態を検査するための動作のループ（306～312）に入る。ステップ306で仮想 GIQ は、その VP がどの動作状態基準を満たしているかを決定する。たとえば、仮想 GIQ は、VP の動作状態に基づいて、所定の1組の基準のうち、どの基準をその VP が満たしているかを示す値（VP_{crit}）を返すルーチンを呼び出すことができる。前述の通り、この基準としては適時に割込みを処理する VP の能力を示すことができる任意の適当な1組の基準を含むことができる。

10

20

30

40

50

、どの基準を満たしているかに基づいて VP_{CRIT} に値を割り当てるために適当なアルゴリズムを使用することができる。

【0030】

たとえば、前述の例示的な基準を使用して、以下の表 I に示す 4 通りの VP_{CRIT} の値のうちの 1 つを VP に割り当てることができる。

【0031】

表 I - VP の基準の例

$VP_{CRIT} = 1$	VP は実行中であり、割込みソースはマスクされていない
$VP_{CRIT} = 2$	VP は実行中であり、割込みソースはマスクされている
$VP_{CRIT} = 3$	VP は実行中ではなく、割込みソースはマスクされていない
$VP_{CRIT} = 4$	VP は実行中ではなく、割込みソースはマスクされている

10

【0032】

表に示すように、1 という VP_{CRIT} の値は最も好ましい動作状態に対応する可能性があり、4 という VP_{CRIT} の値は最も好ましくない動作状態に対応する可能性がある。図 6 は、上記の表 I にリストした特定の 1 組の動作状態基準を適用し、例示した VP_{CRIT} の値を返すための方法を示している。動作 400 は、選択した仮想プロセッサの実行状況と現行割込み優先順位レベルを入手することにより、ステップ 402 から始まる。たとえば、仮想 G1Q は、仮想プロセッサ 114 に関連する 1 組の仮想割込みレジスタ 116 のうちの 1 つから現行割込み優先順位レベルを読み取ることができる。

【0033】

ステップ 404 では、選択した仮想プロセッサが現在実行中であるかどうかについての判定が行われる。選択した仮想プロセッサが実行中である場合、処理はステップ 403 に移行し、そこで選択したプロセッサが仮想割込みソースをマスクしているかどうかについての判定が行われる。選択した仮想プロセッサが仮想割込みソースをマスクしていない場合、ステップ 406 で 1 という基準レベル (VP_{CRIT}) が返される可能性があり、この例では、最も有利な動作状態を示す（たとえば、選択した仮想プロセッサは仮想割込みを直ちに処理できるはずである）。これに反して、選択した仮想プロセッサが仮想割込みソースをマスクしている場合、ステップ 408 で 2 という VP_{CRIT} が返される可能性がある。選択した仮想プロセッサが実行中ではない場合、処理はステップ 410 に移行して、仮想プロセッサが仮想割込みソースをマスクしているかどうかを判定する。マスクしていない場合はステップ 412 で 3 という VP_{CRIT} が返される可能性があり、仮想プロセッサが仮想割込みソースをマスクしている場合はステップ 414 で 4 という VP_{CRIT} が返される。

20

30

【0034】

図 5 に戻って参照すると、ステップ 308 で判定したように選択した仮想プロセッサが最も有利な動作状態（たとえば、 $VP_{CRIT} = 1$ ）を有している場合、現在選択している仮想プロセッサが仮想割込みを直ちに処理できるはずなので、他の仮想プロセッサを検査する必要はない。したがって、ステップ 310 で、選択した仮想プロセッサに仮想割込みが提示される。しかし、選択した仮想プロセッサが最も有利な動作状態以外の動作状態を有する場合（たとえば、仮想プロセッサが実行中ではないかまたは仮想割込みソースをマスクしている場合）、後で比較するためにステップ 312 で動作状態が記録される。次に処理はステップ 304 に戻り、次の仮想プロセッサを選択する。

40

【0035】

どの仮想プロセッサも最も有利な動作状態にない場合、動作はステップ 314 に移行し、そこで他の仮想プロセッサに比べて最も有利な動作状態を有する仮想プロセッサに仮想割込みが提示される。たとえば、仮想 G1Q は単純なルーチンを実行して、ステップ 312 で記録したようにどの仮想プロセッサ 114 が最低の VP_{CRIT} を有するかを識別し、識別した仮想プロセッサ 114 に仮想割込みを提示することができる。複数の仮想プロセッサが同じ最も有利な動作状態 (VP_{CRIT} の値) を有する場合、仮想 G1Q 126 は任意に、いずれか一方に仮想割込みを提示するか、または、たとえば他の適当な基準を考慮してアービトレーションを実行することができる。

50

【0036】

当然のことながら、動作300は、種々の実現例のために様々に修正することができる。たとえば、最も有利な動作状態を有する仮想プロセッサ114に仮想割込みを提示するのではなく、仮想G1Q126は、仮想プロセッサ114が1組の基準のうちの少なくとも1つを有していることを要求することもできる。たとえば、1組の基準のうちのいずれも満たしていない場合、どの仮想プロセッサが最初に仮想割込みを処理できるようになるかを予測するための有効な根拠がまったくない可能性がある。したがって、一部の実施形態では、仮想G1Q126に関連する仮想プロセッサ114のいずれも基準をまったく満たしていない場合（たとえば、いずれも実行中ではなく、いずれについても仮想割込みソースがマスクされている場合）、仮想G1Q126は単に待つだけになる可能性がある。たとえば、仮想G1Q126は単に、関連仮想プロセッサ114のうちの1つが少なくとも1つの基準を満たすまで仮想割込みを保留のままにすることができるだけである。

【0037】

仮想デバイスをサポートし、複数のオペレーティング・システムが1つのロジカル・パーティション・コンピュータ・システム内の物理的プロセッサを共用できるようにするために、仮想割込みを使用する。仮想割込みを処理するために最適の動作状態にある仮想プロセッサに仮想割込みを経路指定することにより、本発明の実施形態は、仮想割込みが適時に処理されることを保証することができる。

【0038】

上記の説明は本発明の実施形態に向けられたものであるが、本発明の基本範囲を逸脱せずに本発明の他の実施形態を考案することができ、その範囲は特許請求の範囲によって決定される。

【0039】

まとめとして、本発明の構成に関して以下の事項を開示する。

【0040】

(1) 仮想割込みを処理するための方法において、複数の仮想プロセッサのうちの少なくとも1つの動作状態に基づいて前記仮想割込みを処理するために前記複数の仮想プロセッサから1つを選択するステップと、

前記選択した仮想プロセッサに前記仮想割込みを提示するステップとを有する方法。

(2) 前記仮想割込みを処理するために前記仮想プロセッサから1つを選択するステップが、

前記仮想プロセッサのうちの少なくとも1つの前記動作状態を検査して、前記少なくとも1つの仮想プロセッサが1つまたは複数の基準を満たしているかどうかを判定するステップを有する、上記(1)に記載の方法。

(3) 前記基準のうちの1つが、前記少なくとも1つの仮想プロセッサが現在実行中であるかどうかを有する、上記(2)に記載の方法。

(4) 前記方法が、

前記仮想割込みのソースを決定するステップをさらに有し、

前記基準のうちの1つが、前記少なくとも1つの仮想プロセッサが前記仮想割込みの前記ソースをマスクしているかどうかを有する、上記(2)に記載の方法。

(5) 前記少なくとも1つの仮想プロセッサが実行中であり、前記仮想割込みソースがマスクされていない場合に、前記少なくとも1つの仮想プロセッサに前記仮想割込みを提示するステップをさらに有する、上記(4)に記載の方法。

(6) 前記少なくとも1つの仮想プロセッサに関連する仮想割込みレジスタに基づいて前記少なくとも1つの仮想プロセッサの前記仮想割込みソースがマスクされているかどうかを判定するステップであって、前記仮想割込みレジスタが前記少なくとも1つの仮想プロセッサの現行割込み優先順位レベルを示すステップをさらに有する、上記(5)に記載の方法。

(7) 仮想割込みを提示するために複数の仮想プロセッサから1つを選択するための方法において、

10

20

30

40

50

前記仮想プロセッサのうちのいずれかが現在実行中であるかどうかを判定するステップと、

実行中である場合にいずれかの実行中の仮想プロセッサの前記仮想割込みのソースがマスクされていないかどうかを判定するステップと、

マスクされている場合に現在実行中であり、前記仮想割込みソースがマスクされていないと判定された仮想プロセッサに前記仮想割込みを提示するステップとを有する方法。

(8) 前記仮想プロセッサのいずれも現在実行中ではなく、前記仮想割込みのソースがマスクされていない場合に、

前記仮想プロセッサのうちのいずれかが現在実行中であるかどうかを判定するステップと、

実行中である場合に現在実行中であると判定された仮想プロセッサに前記仮想割込みを提示するステップとをさらに有する、上記(7)に記載の方法。

(9) 前記仮想プロセッサのいずれも現在実行中ではない場合に、

前記仮想プロセッサのうちのいずれかの前記仮想割込みの前記ソースがマスクされていないかどうかを判定するステップと、

マスクされている場合に前記仮想割込みの前記ソースがマスクされていないと判定された仮想プロセッサに前記仮想割込みを提示するステップとをさらに有する、上記(8)に記載の方法。

(10) 前記仮想プロセッサのいずれも現在実行中ではなく、前記仮想プロセッサのそれぞれの前記割込みの前記ソースがマスクされている場合にのみ、現在実行中ではなく、前記仮想割込みの前記ソースがマスクされている仮想プロセッサに前記仮想割込みを提示するステップをさらに有する、上記(9)に記載の方法。

(11) 仮想割込みを処理するために複数の仮想プロセッサから1つを選択するためのプログラムを含むコンピュータ・プログラムにおいて、プロセッサによって実行されると、前記プログラムが、

前記仮想プロセッサの動作状態に基づいて前記仮想割込みを処理するために前記複数の仮想プロセッサから1つを選択するステップと、前記選択した仮想プロセッサに前記仮想割込みを提示するステップとを有する動作を実行する、コンピュータ・プログラム。

(12) 前記仮想割込みを処理するために前記仮想プロセッサから1つを選択するステップが、

前記仮想プロセッサのうちの少なくとも1つの前記動作状態を検査して、前記少なくとも1つの仮想プロセッサが1つまたは複数の基準を満たしているかどうかを判定するステップを有する、上記(11)に記載のコンピュータ・プログラム。

(13) 前記基準のうちの1つが、前記少なくとも1つの仮想プロセッサが現在実行中であるかどうかを有する、上記(12)に記載のコンピュータ・プログラム。

(14) 前記動作が、

前記仮想割込みのソースを決定するステップをさらに有し、

前記基準のうちの1つが、前記少なくとも1つの仮想プロセッサが前記仮想割込みの前記ソースをマスクしているかどうかを有する、上記(13)に記載のコンピュータ・プログラム。

(15) 前記動作が、前記少なくとも1つの仮想プロセッサが実行中であり、前記仮想割込みソースがマスクされていない場合に、前記少なくとも1つの仮想プロセッサに前記仮想割込みを提示するステップをさらに有し、前記基準のうちの1つが、前記少なくとも1つの仮想プロセッサが現在実行中であるかどうかを有する、上記(12)に記載のコンピュータ・プログラム。

(16) 少なくとも1つのロジカル・パーティションと、

前記ロジカル・パーティションに関連する複数の仮想プロセッサと、

前記ロジカル・パーティションをターゲットにする仮想割込みを受信し、前記仮想プロセッサのうちの少なくとも1つの動作状態に基づいて前記仮想割込みを処理するために前記仮想プロセッサから1つを選択し、前記選択した仮想プロセッサに前記仮想割込みを提

示するように構成される仮想グローバル割込み待ち行列とを有するロジカル・パーティション・コンピュータ・システム。

(17) 現在実行中の仮想プロセッサがまったくない場合にのみ、現在実行中ではない仮想プロセッサに仮想割込みを提示するように前記仮想グローバル割込み待ち行列が構成される、上記(16)に記載のロジカル・パーティション・コンピュータ・システム。

(18) 複数の仮想プロセッサが実行中である場合に、

前記仮想割込みのソースをマスクしていない実行中の仮想プロセッサがまったくない場合にのみ、前記仮想割込みの前記ソースをマスクしている実行中の仮想プロセッサに割込みを提示するように前記仮想グローバル割込み待ち行列が構成される、上記(17)に記載のロジカル・パーティション・コンピュータ・システム。

(19) 仮想割込みを生成可能な1つまたは複数の仮想デバイスをさらに有する、上記(16)に記載のロジカル・パーティション・コンピュータ・システム。

(20) 前記仮想グローバル割込み待ち行列が、複数のロジカル・パーティション間で共用されるハードウェア・グローバル割込み待ち行列を仮想化する、上記(16)に記載のロジカル・パーティション・コンピュータ・システム。

(21) 複数の関連仮想プロセッサを有する少なくとも1つの追加のロジカル・パーティションと、

前記少なくとも1つの追加のロジカル・パーティションに関連する少なくとも1つの追加の仮想グローバル割込み待ち行列とを有する、上記(16)に記載のロジカル・パーティション・コンピュータ・システム。

【図面の簡単な説明】

【0041】

【図1】本発明により例証として使用するロジカル・パーティション・コンピュータ・システムを示す図である。

【図2】本発明の諸実施形態によるコンポーネントの関係図である。

【図3】本発明の諸実施形態によるコンポーネントの関係図である。

【図4】本発明の諸実施形態によるコンポーネントの関係図である。

【図5】本発明の一実施形態により仮想割込みを処理するための例示的な動作を示す流れ図である。

【図6】本発明の一実施形態により仮想プロセッサの動作状態を決定するための例示的な動作を示す流れ図である。

【符号の説明】

【0042】

1 1 0₁ ロジカル・パーティション

1 1 0_N ロジカル・パーティション

1 1 2₁ オペレーティング・システム

1 1 2_N オペレーティング・システム

1 1 4₁ 仮想プロセッサ(複数も可)

1 1 4_N 仮想プロセッサ(複数も可)

1 1 6₁ 仮想割込みレジスタ

1 1 6_N 仮想割込みレジスタ

1 1 8₁ 仮想デバイス

1 1 8_N 仮想デバイス

1 2 0 パーティション・マネージャ

1 2 4 仮想割込み管理機構

1 2 6 仮想グローバル割込み待ち行列(複数も可)

1 3 0₁ 物理的プロセッサ

1 3 0_N 物理的プロセッサ

1 3 6₁ 割込みレジスタ

1 3 6_N 割込みレジスタ

10

20

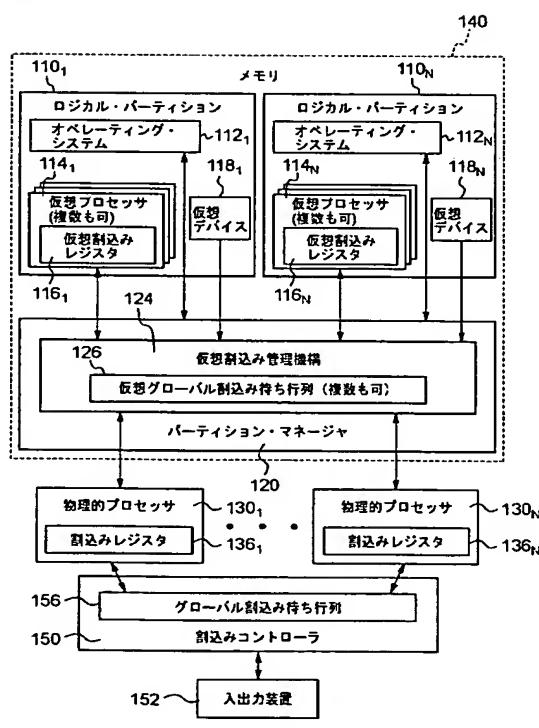
30

40

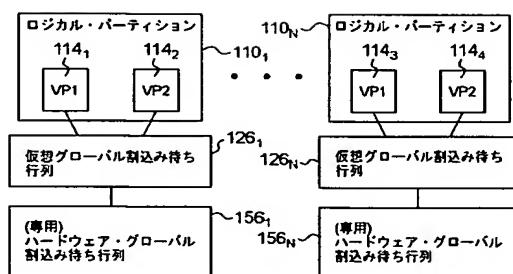
50

140 メモリ
 150 割込みコントローラ
 152 入出力装置
 156 グローバル割込み待ち行列

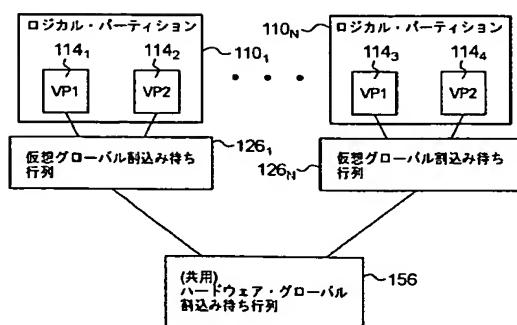
【図 1】



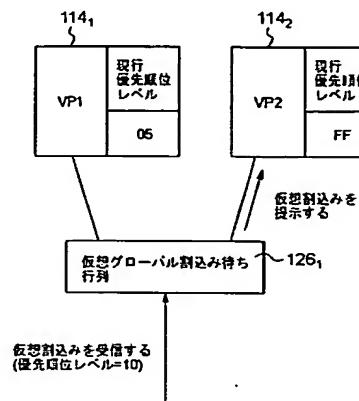
【図 2】



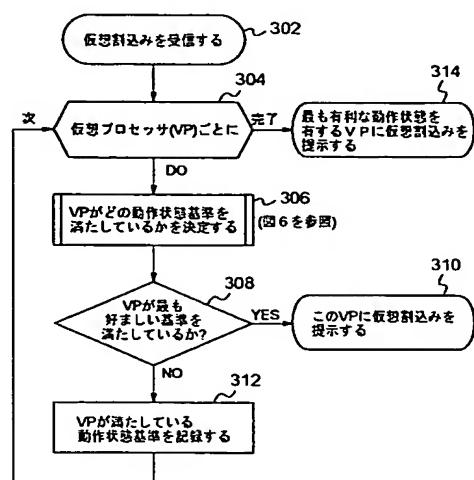
【図 3】



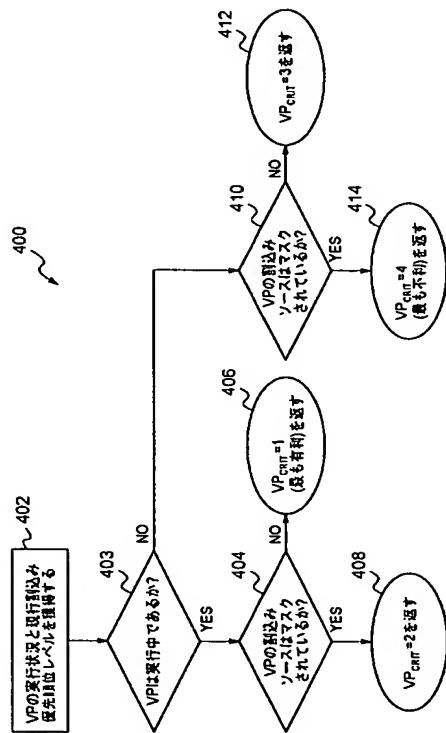
【図4】



【図5】



【図6】



フロントページの続き

(72)発明者 ウィリアム・ジョゼフ・アームストロング
アメリカ合衆国55901 ミネソタ州ロチェスター ニクラウス・ドライブ・エヌ・ダブリュ
5106

(72)発明者 デヴィッド・アンソニー・ラーソン
アメリカ合衆国55901 ミネソタ州ロチェスター 5番通り・エヌ・ダブリュ 4812

(72)発明者 ナレーシュ・ナイヤル
アメリカ合衆国55901 ミネソタ州ロチェスター ベルモラル・レーン・エヌ・ダブリュ 5
233

Fターム(参考) 5B098 BB05 EE01 GA04 HH01 HH04

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.